Dynamic Organization of Search Results Using the UMLS

Wanda Pratt
Section on Medical Informatics
Stanford University, Stanford, CA

When people search the medical literature, they often are overwhelmed by the large number of documents retrieved. Many systems try to solve this problem by helping the user formulate a more specific search strategy. However, when users do not have a more specific question, they need tools to help them explore and understand the results, rather than to eliminate a portion of those results. This paper describes an approach that addresses this need by automatically grouping the results of a broad search into meaningful categories based on the user's query. This approach combines the main benefit of clustering techniques with the main benefit of classification techniques by taking advantage of the domain knowledge present in the UMLS. I present a preliminary evaluation that demonstrates that a categorization produced by this approach corresponds reasonably well to a physician's categorization.

INTRODUCTION

Current information-retrieval tools usually return a simple list of documents that match the specified search criteria. Certain tools order the documents according to relevance criteria, but rarely are the documents grouped in a meaningful way. The returned list of documents is often too long for the user to browse thoroughly, so she is likely to miss many useful documents. Such a list of documents does not provide information about (1) what kinds of information are represented in (or are absent from) the list entries, (2) how the documents relate to the query, or (3) how the documents relate to one another.

I propose that organizing search results will provide this information, thereby helping users to explore the information space related to their query. I am developing an approach that automatically generates a hierarchical organization of document categories and assigns the appropriate documents to each category based on (1) the type of query, (2) the documents retrieved, and (3) a taxonomic model of the domain. I call this approach **dynamic categorization** because it dynamically generates the categorization structure as well as the category labels. The categorization generated by this approach will help users to find specific information efficiently, and to learn more about the information that is available.

This approach will be particularly useful when a user has a general question and is unable to use more specific search criteria. For example, a woman with breast cancer might want to know the latest information on possible complications of a mastectomy. A search in a recent MEDLINE subset on the keywords *mastectomy adverse effects* yields over 350 documents. Without a more specific question, she cannot narrow the search criteria. With only a list of results, she might never form an accurate model of all possible complications by browsing through that list. A tool that categorized the documents according to the adverse effects discussed would help her to see the possibilities, and would enable her to browse the documents easily for the effects that are of most concern to her.

PREVIOUS WORK

Other researchers have explored using medical query models in information retrieval. Cimino and colleagues identified many types of generic queries.[1,2] They encoded expertise from librarians to create specialized search strategies for each generic query. However, they did not use the query information to organize search results. They assumed there were only a few documents relevant to the query, so they used the query information to generate a more specific search. In contrast, my research focuses on queries for which many documents are relevant, such that making the search more specific would reduce the number of relevant documents presented to the user.

To group documents, we can use either manual label assignment or automatic techniques. Manual approaches provide clear category labels; however, they rely on a person to examine the document and to assign one or more category labels to each document. The Yahoo! service, for example, uses this method for categorizing web pages. Such labeling is time consuming and subjective. Furthermore, the person assigning labels needs to think of all ways a user may be interested in the document and must assign all corresponding labels. This approach could result in many labels that do not correspond well to a user's query.

Automatic approaches to grouping documents include clustering and classification. Both techniques usually represent each document as a vector of all words that appear in the documents. They apply a machine-learning algorithm to those vectors to determine the document groupings. The approaches differ in the type of machine-learning algorithm used; **clustering** uses unsupervised-learning algorithms, whereas, **classification** uses supervised-learning algorithms.

1091—8280/97/$5.00

Clustering techniques, such as those reviewed by Willet, try to find hidden structure among the documents.[3] They look for associations among the documents and form the document groups based on those associations. To determine the degree of association among documents, clustering requires a similarity metric. One typical metric is the number of words that the documents have in common. The clustering techniques then label each group (or cluster) with that group's commonly occurring words. Since the algorithms are inferring structures based on word occurrences, the clusters generated are not always meaningful to the user. None of these algorithms use any information about the user's query in forming the clusters, so the groups may not correspond well to the user's query either. Since the document groups are labeled only by the frequently occurring words in the group, the user may not form a good model of the kinds of documents present in the cluster.

In contrast, classification techniques, such as those by Yang and Chute, are given a structure among the documents and must infer the criteria for a document to belong to a group in that structure.[4,5] The structure is provided by a training set that contains a large number of documents assigned to predefined categories. Several such training sets have already been created[6]; however, all were created independent of the user's query. Although the document groups have meaningful labels, they are predefined, so they cannot adapt to the user's query or to the distribution of documents in the search results. For example, if a document in the search results discussed an unexpected complication such as arthritis that was not one of the predefined categories in the training set, classification techniques would not be able to generate a new category for that complication.

DYNAMIC CATEGORIZATION

Dynamic categorization incorporates the main advantages of clustering and classification techniques. The benefit of clustering is that the organization of the documents is influenced by the set of documents being clustered. The benefit of classification is that the documents are organized into meaningful groups that have meaningful labels. Dynamic categorization is based on three key premises: (1) an appropriate categorization depends both on the user's query and on the documents returned from the query, (2) the type of query can provide valuable information about the expected types of categories and about the criteria for assigning documents to those categories, and (3) taxonomic knowledge about terms in the document can enable useful and accurate categorization.

Figure 1 shows an overview of the components for dynamic categorization and their interactions. I have implemented a prototype called DynaCat, which uses dynamic categorization to organize medical search results for patients. DynaCat uses the Oncology Knowledge Server developed by Lexical Technology, Inc. to search the CancerLit document collection and to retrieve information from the terminology model.[7]

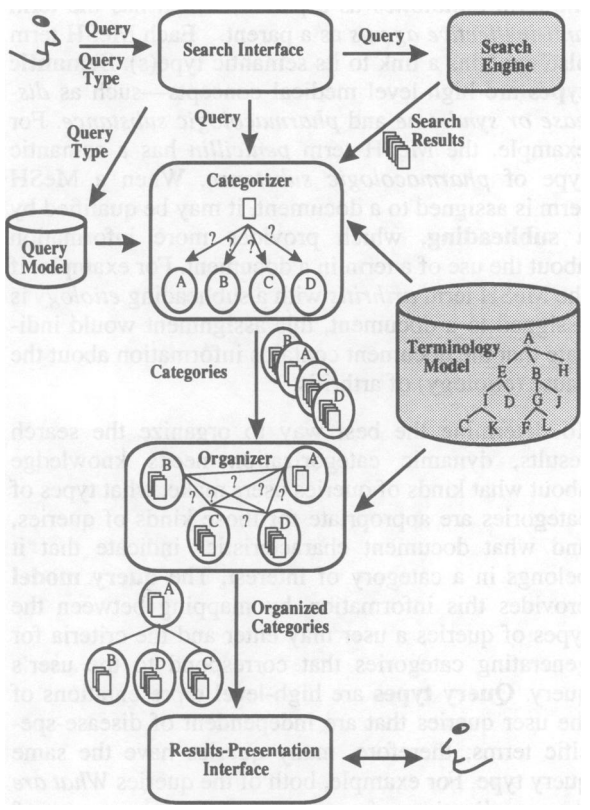

Figure 1: System architecture. The components in white are those that I created. The ones in grey were created by Lexical Technology, Inc., and by the National Library of Medicine. The categorizer uses the categorization criteria from the query model and the search results to assign each document in the search results to one or more categories. The list of categories and the hierarchical placement of those categories in the terminology model are used by the organizer to create a query-sensitive hierarchical categorization of the user's search results.

Dynamic categorization requires two domain models: a terminology model and a query model. The **terminology model** must be a hierarchical model of domain terms, where **terms** may be single words, abbreviations, acronyms, or multiword phrases. For the medical domain, I use the terminology model created by the National Library of Medicine, the Unified Medical Language System (UMLS), which provides information on over 500,000 biomedical terms.[8] I use two parts: the Medical Subject Headings (MeSH) and the semantic types in the UMLS Semantic Network. **MeSH** terms are keywords assigned to medical documents in MEDLINE. MeSH terms are organized in a hierarchy. For example, the MeSH term *penicillin* has

the term *antibiotics* as a parent, which has the term *anti-ineffective agents* as a parent. Each MeSH term also contains a link to its semantic type(s). **Semantic types** are high-level medical concepts—such as *disease or syndrome* and *pharmacologic substance*. For example, the MeSH term *penicillin* has a semantic type of *pharmacologic substance*. When a MeSH term is assigned to a document, it may be qualified by a **subheading**, which provides more information about the use of a term in a document. For example, if the MeSH term *arthritis* with a subheading *etiology* is assigned to a document, this assignment would indicate that the document contains information about the cause (etiology) of arthritis.

To determine the best way to organize the search results, dynamic categorization needs knowledge about what kinds of queries users make, what types of categories are appropriate for those kinds of queries, and what document characteristics indicate that it belongs in a category of interest. The **query model** provides this information by mapping between the types of queries a user may enter and the criteria for generating categories that correspond to the user's query. **Query types** are high-level representations of the user queries that are independent of disease-specific terms; therefore, many queries have the same query type. For example, both of the queries *What are the complications of a mastectomy for breast cancer?* and *What are the side effects of taking the drug Seldane to treat allergies?* have the same query type of *treatment—adverse effects*, even though they mention different diseases and different treatments. The types of queries are organized in a hierarchy that represents the intersection of the kinds of medical information that are available in the medical literature and the kinds of questions that users typically ask. Each query type is mapped to the (1) **categorization criteria**, which specify the conditions that must be satisfied for a document to belong to that type of category, and (2) a **label generator**, which provides the procedure for creation of a category label. Currently, I represent the categorization criteria as a taxonomic constraint, which is the list of allowed semantic types for the document's MeSH terms and the allowed subheadings. For now, the label generator is a simple function that returns the name of the MeSH term that satisfies the categorization criteria. In the future, it may be necessary to have user-specific label generators to allow for vocabulary differences between patients and clinicians.

To generate a categorization for the results of a search, the **categorizer** needs to determine which topics discussed in the search results correspond to topics of interest for the given query type. To accomplish this task, the categorizer retrieves the categorization criteria for the corresponding query type, and examines each document in the set of results individually. For each document, it checks the categorization criteria of every category type in the categorization criteria and uses the UMLS Semantic Network to look up the semantic type for each of that document's MeSH terms. When a MeSH term's semantic type matches any of the semantic types and subheadings in the categorization criteria for a category type, the categorizer adds the document to the category labeled with that MeSH term. If such a category has not already been created, a new category is generated by the label generator. Every MeSH term in a document is checked against the categorization criteria. Therefore, each document may be categorized under more than one label.

For example, if the query was *What are the adverse effects of a mastectomy?*, the query type would be *treatment: adverse effects*. One of the corresponding categorization criteria would specify a semantic type of *disease or syndrome* and a subheading of *etiology*. Then, if a document was indexed by the MeSH terms *lymphedema-etiology, arthritis-etiology, diagnostic imaging, mastectomy-adverse effects,* and *middle age*, DynaCat would categorize it under both *lymphedema* and *arthritis*. The document would not be categorized under *diagnostic imaging, mastectomy,* or *middle age* because those terms do not satisfy the categorization criteria. Note that the terms *lymphedema* and *arthritis* were not predefined category labels in the query model. Rather, they were generated dynamically as category labels because they satisfied criteria in the query model.

The goal of the category **organizer** is to create a hierarchical organization of the categories that is not too broad or deep, as defined by set thresholds. The organizer produces the final categorization hierarchy based on the distribution of documents from the search results. When there are many categories at one level in the hierarchy, the categories are grouped under a more general label. DynaCat generates the more general label by traversing up the MeSH term hierarchy to find a term that is a parent to several document categories.

The **results-presentation interface** creates a frame-based web document to present the hierarchical organization of categories to the user. Figure 2 illustrates the web document that corresponds to the search on the adverse effects of a mastectomy.

PRELIMINARY EVALUATION

The goal of my preliminary evaluation was to determine the **accuracy** of the categorization, defined as how well DynaCat generates a reasonable categorization hierarchy and places the documents in all and only the appropriate categories. I evaluated DynaCat using the query: *What are the complications of a mastectomy for breast cancer?* Figure 2 shows the web

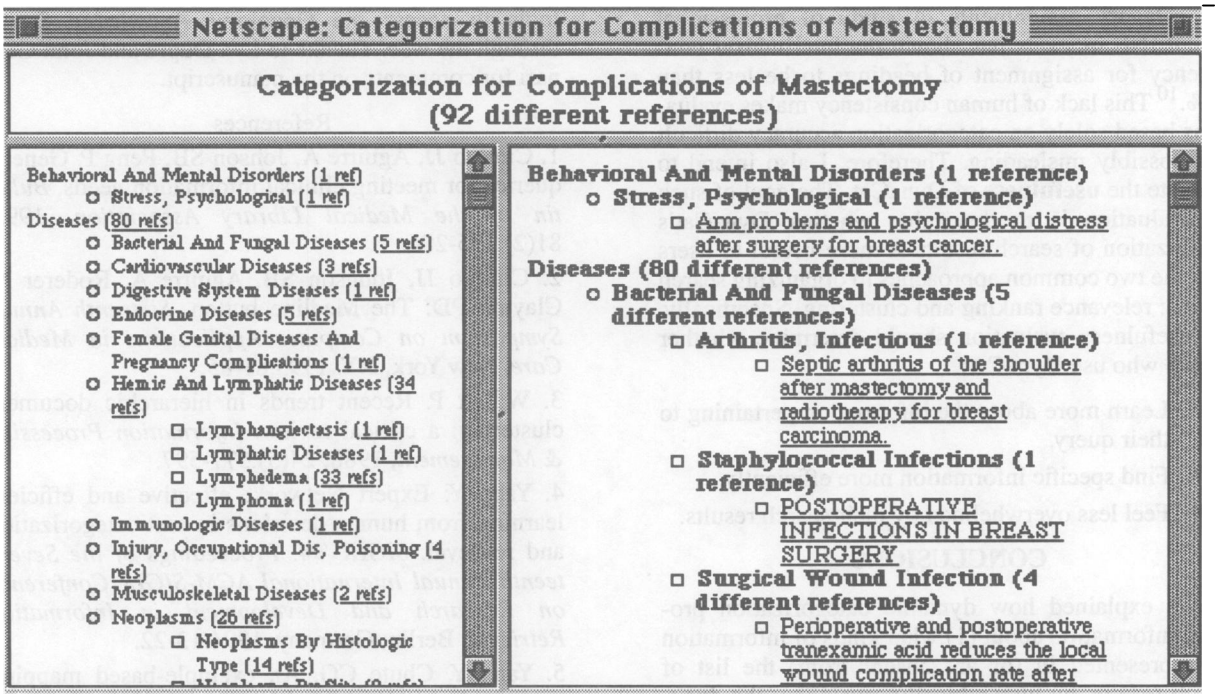

Figure 2: Dynamic categorization for a query on the adverse effects of a mastectomy. The document is split into three frames: one horizontal frame or row along the top, and two vertical frames or columns. The top frame always contains the query and the number of different citations that satisfied the query. The left frame contains the most general categories. This frame is designed to be used like a table of contents for a book. The numbers in parentheses indicate the number of unique citations or references in the named category and provide a hyperlink to the corresponding category as it appears in the entire categorization structure. The right frame can contain either the entire hierarchical organization of categories with the titles of the citations that belong to each category, or the entire citation. The citation's title in the categorization hierarchy is a hyperlink to the entire citation, including the document's title, author, source, type, language, unique identifier, subject headings, and abstract.

page that DynaCat generated for that query. A search for *Mastectomy Adverse Effects* using the Oncology Knowledge Server resulted in 92 different citations from CancerLit. The number of categories that DynaCat generated in the initial categorization was 53. If DynaCat had categorized the citations using every MeSH term of every citation in the search results, there would have been 263 categories. In the hierarchical organization of categories, DynaCat generated 35 more categories for a total of 88.

To measure accuracy, I compared the categorization generated by DynaCat to a physician's categorization. I randomly selected 30 citations from the original 92 search results and asked a physician to assign each citation to one or more categories in the hierarchy of categories generated by DynaCat.

For each category, I determined the precision and recall of DynaCat using the physician as the gold standard. **Precision** was calculated as the number of citations that both the physician and DynaCat assigned to the category divided by the number of citations DynaCat assigned to the category. **Recall** was calculated as the number of citations that both the physician and DynaCat assigned to the category divided by the number of citations that the physician assigned to the category.

The precision of DynaCat, averaged across all categories, was 70.2%. The average recall was 44.0%. Since there are no other systems that perform this exact task, it is difficult to use these figures in any comparison. However, a related task is the automatic classification of MEDLINE documents into MeSH terms. Yang and Chute evaluated several automatic classification approaches; they found the best approach yielded an average precision of 34.9%.[9] Since the number of categories for the dynamic-categorization task is much fewer than that for the task of assigning MeSH terms to documents, DynaCat should provide a categorization with higher average precision. However, since DynaCat had about twice the average precision, this evaluation provides evidence that DynaCat is as accurate as other automatic categorization approaches.

A problem with any approach to categorization is the subjective nature of the task. People disagree about

the category assignments. A study of professional MEDLINE indexers has shown the interindexer consistency for assignment of headings to be less than 49%.[10] This lack of human consistency makes evaluations based solely on categorization accuracy difficult and possibly misleading. Therefore, I also intend to evaluate the **usefulness** of DynaCat. The goal of such an evaluation is to determine whether DynaCat's organization of search results is more useful to users than the two common approaches to organizing search results: relevance ranking and clustering. Specifically, the usefulness evaluation should determine whether people who use DynaCat:

- Learn more about the information pertaining to their query.
- Find specific information more efficiently.
- Feel less overwhelmed by their search results.

CONCLUSIONS

I have explained how dynamic categorization provides information about (1) what kinds of information are represented in (or are absent from) the list of search results, by hierarchically organizing the document categories and by providing meaningful labels for each category; (2) how the documents relate to the query, by making the categorization structure dependent on the type of query; and (3) how the documents relate to one another, by grouping documents that cover the same topic into the same category. Although my current approach requires preassigned keywords (e.g., MeSH terms), it could be expanded to cover documents without keywords by using one of several experimental systems that automatically assign keywords to documents, such as the system by Fowler and colleagues, which assigns MeSH labels to web documents.[11]

My preliminary evaluation demonstrated that DynaCat can generate a categorization that corresponds well to a physician's categorization. By providing a useful organization of medical search results, DynaCat will help people to explore the medical literature. It can help patients to use the primary medical literature to become informed about the options available to them in their medical care. It also can assist clinicians to make more efficient and effective explorations of the literature.

Acknowledgments

This work was supported by the NLM grant LM-07033 and the NCI contract N44-CO-61025. Computing facilities were provided by the CAMIS Resource, LM-05305. I thank my advisors, Larry Fagan and Marti Hearst, for helping me to formulate this research. I also thank the people at Lexical Technology, Inc.—particularly Kevin Keck—for providing tools to access the Oncology Knowledge Server through the web. Thanks to Lyn Dupré and John Gennari for comments on the manuscript.

References

1. Cimino JJ, Aguirre A, Johson SB, Peng P. Generic queries for meeting clinical information needs. *Bulletin of the Medical Library Association*, 1993; 81(2):195-206.

2. Cimino JJ, Johnson SB, Aguirre A, Roderer N, Clayton PD: The Medline button. *Sixteenth Annual Symposium on Computer Applications in Medical Care*, New York, NY. 1993:81-85.

3. Willett P. Recent trends in hierarchic document clustering: a critical review. *Information Processing & Management*, 1988; 24(5):577-597.

4. Yang Y: Expert Network: effective and efficient learning from human decisions in text categorization and retrieval. *SIGIR '94. Proceedings of the Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, Berlin, Germany. 1994:13-22.

5. Yang Y, Chute CG. An example-based mapping method for text categorization and retrieval. *ACM Transactions on Information Systems*, 1994; 12(3): 252-277.

6. Hersh W, Buckley C, Leone TJ, Hickam D: OHSUMED: an interactive retrieval evaluation and new large test collection for research. *Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, Berlin, Germany. 1994:192-201.

7. Tuttle MS, Sherertz DD, Fagan LM, Carlson RW, Cole WG, Schipma PB, et al.: Toward an interim standard for patient-centered knowledge-access. *Seventeenth Annual Symposium on Computer Applications in Medical Care (SCAMC)*, New York, NY. 1994: 564-568.

8. McCray AT, Aronson AR, Browne AC, Rindflesch TC, Razi A, Srinivasan S. UMLS knowledge for biomedical language processing. *Bulletin of the Medical Library Association*, 1993; 81(2):184-194.

9. Yang Y, Chute CG: An application of expert network to clinical classification and MEDLINE indexing. *Eighteenth Annual Symposium on Computer Applications in Medical Care (SCAMC)*, Washington, DC. 1994:157-161.

10. Funk ME, Reid CA. Indexing consistency in MEDLINE. *Bulletin of the Medical Library Association*, 1983; 71(2):176-183.

11. Fowler J, Maram S, Kouramajian V, Devadhar V: Automated MeSH Indexing of the World-Wide Web. *Nineteenth Annual Symposium on Computer Applications in Medical Care (SCAMC).*, 1995:893 -897.